

Enhanced speed regulation using separate P and I gain controllers in a fuzzy-PI framework

Minh Duc Pham, Duong Nguyen Trong Qui, Truong Phuoc Hoa

Power Electronics Research Laboratory, Faculty of Electrical and Electronics Engineering,

Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam

Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam

Article Info

Article history:

Received Apr 16, 2025

Revised Oct 4, 2025

Accepted Oct 17, 2025

Keywords:

BLDC motor

Fuzzy-PI control

Separate P and I gain
controllers

Speed reference variation

Speed regulation

ABSTRACT

This paper explores an enhanced method for regulating the speed of brushless DC (BLDC) motors using field-oriented control. Conventionally, a proportional-integral (PI) controller is employed to adjust output speed and current FOC method. While the PI controller is effective in many scenarios, it exhibits limitations including poor performance when the speed reference changes rapidly. To address these limitations, a fuzzy-PI control scheme is proposed in this study with the aim of improving the speed control performance of BLDC motors, especially under rapidly changing speed reference. The proposed two separate fuzzy logic controllers adaptively adjust the proportional and integral gains so that it combines the robustness of fuzzy logic with the steady-state error of PI control. Simulation and experimental results demonstrate that the fuzzy-PI control significantly outperforms the conventional PI controller in terms of BLDC stability, response time, and accuracy. The proposed approach ensures more reliable and efficient speed regulation for BLDC motors, making it a reliable solution for applications where speed reference fluctuate frequently.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Truong Phuoc Hoa

Power Electronics Research Laboratory, Faculty of Electrical and Electronics Engineering

Ho Chi Minh City University of Technology (HCMUT) VNU-HCM

Ho Chi Minh City, Vietnam

Email: tphoa@hcmut.edu.vn

1. INTRODUCTION

The era of industrialization made significant contributions by inventing electric motors. In particular, carbon-brushed direct current (DC) electric motors are commonly used with some disadvantages concerning maintenance costs and no application in hazardous areas [1]. As a result, brushless motors (BLDC) were developed to address these issues and have become an increasingly critical technology for electric vehicles and industrial applications. Controlling a BLDC motor is more difficult than a DC motor because it requires precise synchronization of the stator electromagnetic field with the rotor position, thus requiring additional recognition of the motor speed and position [2].

BLDC motors are used in many different applications, including household appliances such as washing machines, dryers, and compressors, or equipment in the automotive industry such as electronic steering systems, fuel pumps, electric vehicle engines, and robotic arms [3], [4]. From the above applications, in many cases, due to differences in motor load, the speed will fluctuate in a large range. Then, the need to control the speed and dynamic characteristics of the motor is necessary to serve these applications [5], [6].

A classical controller, such as proportional and hysteresis controllers, works in the closed-loop algorithm based on feedback using measurement devices [7]-[9]. But these conventional controllers are not effective with variation in speed, sensitivity towards parameter changes, and non-linear systems. Also, they often suffer from poor disturbance rejection and have the potential to have a long response time. Therefore, it is needed to research an improved control algorithm for the BLDC motor.

Currently, proportional-integral (PI) controllers are being widely applied in many technical fields thanks to their simplicity, durability, and the ability to easily adjust control parameters [10]. However, PI controllers also have significant disadvantages. Controller parameter selection methods such as the Ziegler-Nichols law may not guarantee high-speed performance and response [11]. In addition, conventional PI controllers have fixed parameters, leading to the system being easily unstable to external influences and lacking adaptability [12]. The authors in [13] present a third-harmonic voltage combined with a PLL-based position estimator to achieve a wide-speed control range, but it requires a start-up routine at very low speed and careful PLL tuning.

Another study presents a fuzzy speed controller integrated with a front-end PFC Cuk converter [14]. Nevertheless, it emphasizes power-quality conditioning and reports primarily simulation results without an inner dq-FOC speed control loop. These methods still have some issues, such as reliance on fixed PI gains and a lack of hardware validation under different speed commands. Some conventional methods have been presented to improve BLDC motor control performance. For example, genetic algorithms (GA) have been utilized for optimizing control parameters, but they require offline evaluation and are hard to implement in real-time applications [15], [16]. The sliding mode control (SMC) method provides robustness against parameter variations, but it often introduces chattering, which can cause motor noise [17]. Similarly, H-infinity controllers provide stability under disturbances but it often results in high-order controllers and requires careful selection of weighting functions [18]. These methods still face issues such as the need for detailed models and tuning expertise, which adds implementation complexity and potential incompatibility with existing fixed-gain PI systems.

For torque ripple, switching-table torque control with multi-slope vector selection reduces ripple without time-duration optimization, thus avoiding the high computational cost of predictive/hybrid controllers [19]. Another study presents fuzzy-PI tuned by PSO improves speed transient and damping compared with fixed-gain PI, when integral of time-weighted absolute error is used as an objective [20]. Nevertheless, the multi-slope vector selection is sensitive to torque/flux estimation error, and PSO-tuned fuzzy-PI typically relies on offline optimization and a fixed rule base, so its performance can degrade under shifted operating point. Additionally, model predictive control (MPC) offers advanced control by predicting future motor behavior. However, it requires precise and prior knowledge of the system model and parameters, making it less effective in systems with high uncertainty or variability [21], [22].

In this paper, the separated P and I gain controllers in a fuzzy-PI framework are proposed for several key advantages. They do not require an accurate system model, making it suitable for the nonlinear characteristics of BLDC motors [23]. In addition, the proposed approach uses two fuzzy logic controllers instead of one controller to adaptively adjust the proportional and integral gains based on the error and its derivative. The use of two separate fuzzy controller enhances the controller response and stability. In industrial working conditions, the BLDC motor usually operates under significant switching noise, electromagnetic interference, and back EMF fluctuations, which reduce the control accuracy. In addition to the adaptive fuzzy-PI control, this study adopts a median filter to filter out the BLDC motor current signal from pulse noise or spikes. This enhancement ensures a cleaner current signal that improves accurate and control responses in noisy industrial conditions.

The feasibility of the fuzzy proposed approach is evaluated through MATLAB-Simulink simulations under various conditions. Noise and spikes are also emulated in the simulations to assess the system noise rejection performance, demonstrating the effectiveness of the separated P and I gain controllers in a fuzzy-PI framework with a median filter. In addition, experimental validation is carried out using BLDC prototype, Texas Instruments TMS320F28379D microcontroller. A series of experimental scenarios are examined to evaluate the control performance when reference speeds are changed abruptly.

2. MATHEMATICAL MODEL OF BLDC

2.1. BLDC mathematics model

In this section, the mathematical model of the BLDC motor and its nonlinearity are examined. As depicted in the cross-sectional view of a BLDC motor in Figure 1, AX, BY, and CZ represent the distinct pole pairs. Figure 2 illustrates the relationship among the phase voltage equations of each winding. Because the three phases are identical, for simplification, only phase A is analyzed. The phase-to-ground voltage in phase A can be expressed as:

$$u_A = u_{AN} = R_A i_A + e_{\psi_A}, \quad (1)$$

Where u_A is phase voltage; i_A is phase current; e_{ψ_A} is phase-induced EMF, and R_A is phase resistance. The flux e_{ψ_A} in (1) can be written as (2):

$$\begin{cases} e_{\psi_A} = \frac{d\psi_A}{dt} \\ \psi_A = L_A i_A + M_{AB} i_B + M_{AC} i_C + \psi_{pm}(\theta) \end{cases}, \quad (2)$$

where ψ_{pm} is permanent magnet (PM) flux linkage of phase A; θ is the position angle of the rotor; L_A is the self-inductance of phase A; and M_{AB}, M_{AC} are mutual inductance of phase A with phase B and phase C. The magnitude of $\psi_{pm}(\theta)$ depends on the magnetic field distribution of PM in the air gap. When the rotor rotates anticlockwise, the effective flux in phase A changes relative to the rotor position. Then, substituting (2) into (1) results in (3).

$$u_A = R i_A + \frac{d}{dt} (L_A i_A + M_{AB} i_B + M_{AC} i_C) + e_A \quad (3)$$

Where e_A represents the back electromotive force (back-EMF) for phase A in a PMSM. Because of the balanced three phases, the mutual induction of the three phases is the same $M_{AB} = M_{AC} = M_{BC} = M$. Because the stator has a Y-connected, the relationship of phase current is established $i_B + i_C = -i_A$, and the (3) is then simplified as:

$$u_A = R i_A + (L - M) \frac{di_A}{dt} + e_A \quad (4)$$

The back-EMF e_A is also defined according to [24]:

$$e_A = 2NS\omega B_m f_A(\theta) = \omega \psi_m f_A(\theta) \quad (5)$$

Where B_m and ψ_m are the maximum values of PM density distribution in air gap and flux linkage of each winding, $\psi_m = 2NSB_m$; $f_A(\theta)$ is back-EMF waveform function of phase A.

Extending from phase A to three phases, the matrix form of phase-to-phase voltage in (6) can be obtained as:

$$\begin{bmatrix} u_{AB} \\ u_{BC} \\ u_{CA} \end{bmatrix} = \begin{bmatrix} R & -R & 0 \\ 0 & R & -R \\ -R & 0 & R \end{bmatrix} \begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix} + \begin{bmatrix} L - M & M - L & 0 \\ 0 & L - M & M - L \\ M - L & 0 & L - M \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix} + \begin{bmatrix} e_A - e_B \\ e_B - e_C \\ e_C - e_A \end{bmatrix} \quad (6)$$

From the matrix in (6), the simple circuit including resistance, inductance, and back EMF is shown in Figure 2. In Figure 2, it can be seen that the stator current is the possible control variable in the BLDC system.

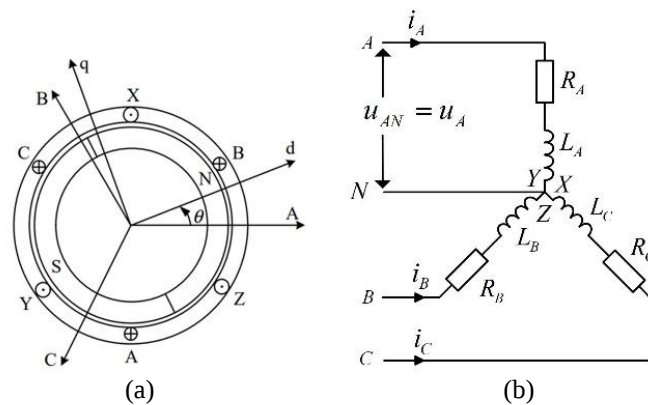


Figure 1. BLDC cross-sectional view and schematic: (a) cross-sectional view and (b) BLDC schematic diagram

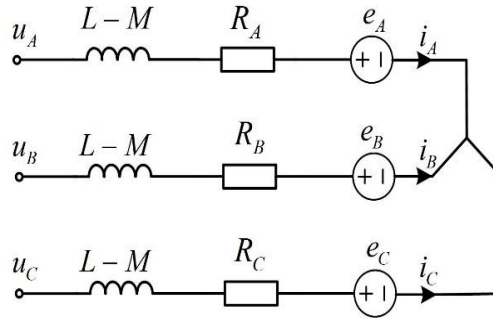


Figure 2. The simplified equivalent circuit of BLDC motor

2.2. Dynamic BLDC mathematics model

BLDC motor performance, like DC motors, can be assessed via energy transfer. Power from the source is mainly transferred to the rotor electromagnetically, generating torque, with some losses in copper and iron. This electromagnetic power equals the sum of each phase current multiplied by its back-EMF, which is calculated as (7).

$$P_e = e_A i_A + e_B i_B + e_C i_C \quad (7)$$

Ignoring the mechanical loss and stray loss, the electromagnetic power is totally turned into kinetic energy and can be represented as:

$$T_e = \frac{P_e}{\omega_m} = \frac{e_A i_A + e_B i_B + e_C i_C}{\omega_m}, \quad (8)$$

where T_e represents electromagnetic torque and ω_m represents angular velocity of rotation. Substituting in (5) into (8) with p is pole pairs, the torque (9) can be represented as:

$$T_e = p[\psi_m f_A(\theta) i_A + \psi_m f_B(\theta) i_B + \psi_m f_C(\theta) i_C], \quad (9)$$

To fully describe the electromechanical system mathematically in Figure 3, the motion equation is established:

$$T_e - T_L = J \frac{d\omega_m}{dt} + B_v \omega_m, \quad (10)$$

where T_L is the load torque; J is rotor moment of inertia and B_v is viscous friction coefficient.

From (10), the phase currents i_A , i_B , i_C change, the electromagnetic torque T_e will vary and changes in the motor speed. In different operating conditions where the moment of inertia J and the viscous friction coefficient B_v exhibit nonlinear behavior, any change in T_e must adapt to reflect these nonlinear characteristics. This highlights the limitation of conventional linear controllers with fixed gain as they struggle to manage the BLDC speed effectively under complex and varied conditions.

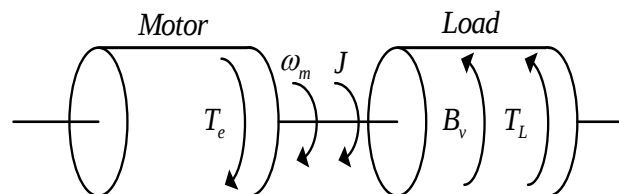


Figure 3. Simplified rotational system with load

3. INNER FIELD ORIENTED CONTROL LOOP AND THE OUTER CONVENTIONAL SPEED CONTROL BASED ON A LINEAR CONTROLLER

3.1. Inner FOC control loop

Transforming the currents into the dq frame simplifies control without losing the original system characteristics. This transformation decouples the flux and torque components, corresponding to the stator currents along the d-axis and q-axis, respectively, which makes managing the system easier by allowing for separate control of the flux and torque components [25]. In (11) shows how the electromagnetic torque T_e is controlled in the dq frame.

$$T_e = \frac{3}{2}p[\psi_m i_q + (L_d - L_q)i_q i_d] \quad (11)$$

In (11), the torque depends on the stator flux linkage (ψ_m) and the current along the q-axis (i_q), with a coupling term involving the d-axis current (i_d), and p is number of pole pairs. In the inner FOC control loop, the q-axis current (i_q) is controlled to regulate the electromagnetic torque, while d-axis current (i_d) is usually set to zero to minimize losses and simplify control. By regulating i_d to ($i_{dref} = 0$), the (11) becomes $T_e = \frac{3}{2}p\psi_m i_q$. The inner FOC control equation for regulating the currents in the d-q reference frame is expressed in (12):

$$\begin{cases} V_{adj_d} = (0 - i_d)K_{pd} + \int K_{id} (0 - i_d)dt \\ V_{adj_q} = (i_{qref} - i_q)K_{pq} + \int K_{iq} (i_{qref} - i_q)dt \end{cases} \quad (12)$$

where i_{dref} is set at 0, and K_{pd} and K_{pq} are control gains, respectively. The inner FOC control loop regulates i_d and i_q through the corresponding dq voltages V_{adj_d} and V_{adj_q} . They are fed into the space vector Pulse Width Modulation (SVPWM) block to generate the pulse signals for the BLDC inverter driver [26].

3.2. Outer conventional speed control based on a linear controller

By using the FOC control loop, the relationship between i_q and motor speed ω_m is expressed as (13).

$$i_q = \frac{2}{3p\psi_m} \left(J \frac{d\omega_m}{dt} + B_v \omega_m + T_L \right) \quad (13)$$

This (13) shows that the q-axis current (i_q) in stator can be used to control the mechanical motor speed. By controlling i_q , the controller can directly regulate the motor speed (ω_m). The conventional PI controller in the speed control loop is designed to minimize the error between the reference speed (ω_{ref}) and the actual speed (ω_m). The controller in (14) is defined as (14).

$$i_{q_ref} = K_p e_\omega + K_i \int e_\omega dt \quad (14)$$

where ($e_\omega = \omega_{ref} - \omega_m$). The conventional PI controller can be represented in both the S-domain and the Z-domain, as (15).

$$\begin{cases} i_{q_ref}(s) = K_p + \frac{K_i}{s}; sdomain \\ i_{q_ref}(z) = K_p + K_i \frac{T_s z + 1}{2 z - 1}; zdomain \end{cases} \quad (15)$$

The PI controller in S domain is used for continuous-time systems to analyze frequency response and stability, and the Z-domain is applied for discrete-time microcontroller with sampling period T_s . The conventional BLDC control block diagram is shown in Figure 4.

Although PI controllers are simple with small computation burden, the conventional PI controller has some limitations. First, the fixed proportional (K_p) and integral (K_i) gains are typically tuned for specific operating speed condition. This makes it less effective under varying loads or speed references, which may cause high speed overshoot and undershoot. Additionally, the fixed I controller may not be sensitive to external disturbances, which increase the settling times when the speed reference suddenly changes. Furthermore, i_d may not be ideal ($i_d = 0$) due to noise in measurements and limited controller bandwidth, so that the system characteristics become nonlinear. These limitations highlight the need for adaptive control strategies, such as the fuzzy-PI controller, to improve performance under dynamic conditions and disturbances.

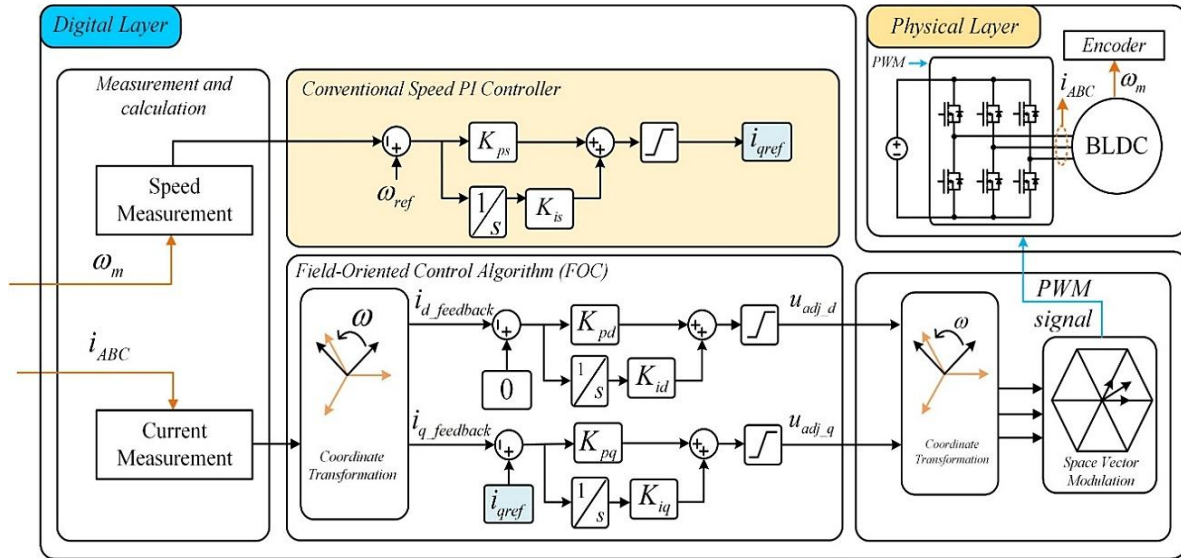


Figure 4. Conventional PI controller for speed regulation

4. PROPOSED ENHANCED SPEED REGULATION USING SEPARATE P AND I GAIN CONTROLLERS IN FUZZY-PI FRAMEWORK

4.1. Basic idea of proposed enhanced speed regulation

To overcome the conventional PI issue, the proposed method utilizes two fuzzy logic controllers (FLCs), which is known for their robustness to address the system uncertainties and nonlinear characteristics. FLC does not require an accurate mathematical model of the system, so it is suitable for BLDC motor drive with uncertainties and nonlinear characteristics. The fuzzy controller rule-based system allows it to continuously adapt the speed error, and its derivative based on the real-time feedback. Instead of using a single fuzzy controller to adjust both proportional and integral gains simultaneously, this approach employs two separate fuzzy logic controllers for adaptively adjusting the proportional (K_p) and integral (K_i) gains.

4.2. The adoption of median filter in BLDC current filtering

Additionally, the median filter is integrated at the input of the fuzzy logic controller to enhance the noise rejection. A sinusoidal wave with an amplitude of 1 and 200 samples per period serves as a representative example for a feedback current signal. Function pulsed noise is added introducing impulsive noise into a signal at a single time step. To emulate the pulsed noise, the function generates a random number between 0 and 1. If this number exceeds the specified threshold (random $\geq$ threshold; threshold = 0.7), a pulse of magnitude (pulse value = 5) is added to the signal. The polarity of the pulse (positive or negative) is also randomly determined. Otherwise, no noise is added. The emulated current signal is then calculated as:

$$y = \text{signal} + \eta; \eta = \text{pulse_value} = \pm 5 \quad (16)$$

The function returns the potentially noise-corrupted signal value (y) for the current time step. The study utilizes a real-time median filter with a moving window to smooth time-series data, and data within the window is then sorted (sortedData). For an odd-sized window (n), the median is the middle element:

$$\text{Med} = \text{sortedData}\left(\frac{n+1}{2}\right) \quad (17)$$

For an even-sized window (n), the median is the middle element:

$$\text{Med} = \frac{\left(\text{sortedData}\left(\frac{n}{2}\right) + \text{sortedData}\left(\frac{n}{2} + 1\right)\right)}{2} \quad (18)$$

The signal with noise and after applying median filter is shown in Figure 5 using Algorithm 1. The blue waveform represents the noisy signal, characterized by pulsed noise, while the red waveform represents the signal after applying the median filter. The red signal after filtering has significantly removed the pulsed noise and has a negligible phase delay, thereby improving the signal quality to feed into two separate fuzzy controllers.

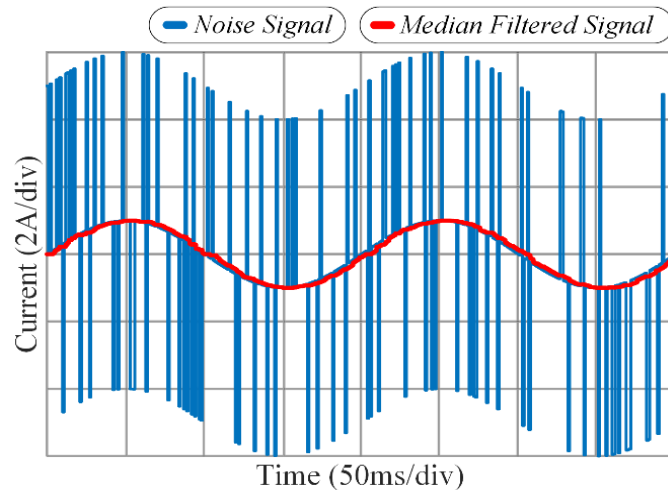


Figure 5. Signal with noise and after applying median filter

Algorithm 1. Algorithm for Median filter for pulsed signal filtering

Input: Noise Current Signal, Window Size

Output: Median Filtered Current Signal

% Shift data within the window

windowData = [windowData(2:end), inputData];

% Apply the median filter

sortedData = sort(windowData);

n = length(windowData);

if mod(n, 2) == 0

output = (sortedData(n/2) + sortedData(n/2 + 1)) / 2;

else

output = sortedData((n + 1) / 2);

end

4.3. The design of a fuzzy logic controller

The control variables fuzzy(P) and fuzzy(I) are outputs of two fuzzy logic controllers, allowing the adjustment of the parameters of the PI controller. The input coefficient is normalized to the scale of [-1; 1] and becomes a crisp input of FLC. Based on experience, the outputs fuzzy(P) and fuzzy(I) with different gain coefficients in a range of 0.4 to 1 and 0.5 to 1. The fuzzy sets are defined as follows: speed error $e = \{NB, NS, ZE, PS, PB\}$ and the derivative of the error $de = \{NB, NS, ZE, PS, PB\}$. NB, NS, ZE, PS, and PB represent negative big, negative small, zero, positive small, and positive big. The output of both fuzzy controllers is similar, written as $fuzzy(P) = fuzzy(I) = \{S, M, B, VB\}$, where S, MS, M, B, and VB represent small, medium, medium, big, and very big. The fuzzy rule tables are provided in Table 1, and the membership function for input and output is shown in Figure 6.

The control surfaces shown in Figures 7(a) and 7(b) represent the nonlinear behavior of the fuzzy logic controllers compared to traditional PI controllers. Unlike the linear control response of a PI controller, fuzzy control surfaces exhibit a more complex, nonlinear relationship between the input variables (error and derivative of error) and the control output. This allows the system to dynamically adjust the controller gains based on real-time inputs, which enhances the flexibility and adaptability of the system.

The advantage of the fuzzy control surface over a conventional PI controller lies in its ability to handle uncertainty and nonlinearity in the system. While PI controllers use fixed proportional and integral gains, the fuzzy logic controller can modify these gains according to the operating conditions, leading to better performance in systems with varying dynamics or external disturbances. Additionally, the fuzzy controllers can operate over a wide range of input conditions, ensuring robust performance across different scenarios. This results in faster response times, reduced overshoot, and better stability compared to conventional PI controllers, which are often limited by their fixed-gain nature. The combined proposed BLDC control block diagram is shown in Figure 8.

Table 1. Rule-based for P and I gains

Rule-base for P gain regulation						Rule-base for I gain regulation					
$e \backslash de$	NB	NS	Z	PS	PB	$e \backslash de$	NB	NS	Z	PS	PB
NB	VB	VB	Z	PS	PB	NB	VB	VB	Z	PS	PB
NS	VB	B	Z	PS	PB	NS	VB	VB	Z	PS	PB
Z	B	S	Z	PS	PB	Z	M	M	Z	PS	PB
PS	S	VB	Z	PS	PB	PS	S	M	Z	PS	PB
PB	VB	VB	Z	PS	PB	PB	VB	B	Z	PS	PB

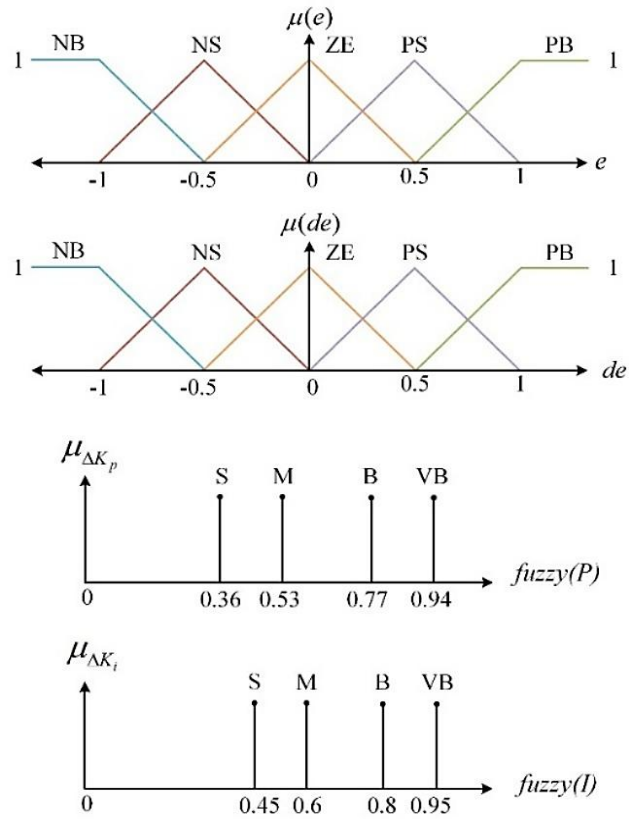


Figure 6. The membership function for input and output

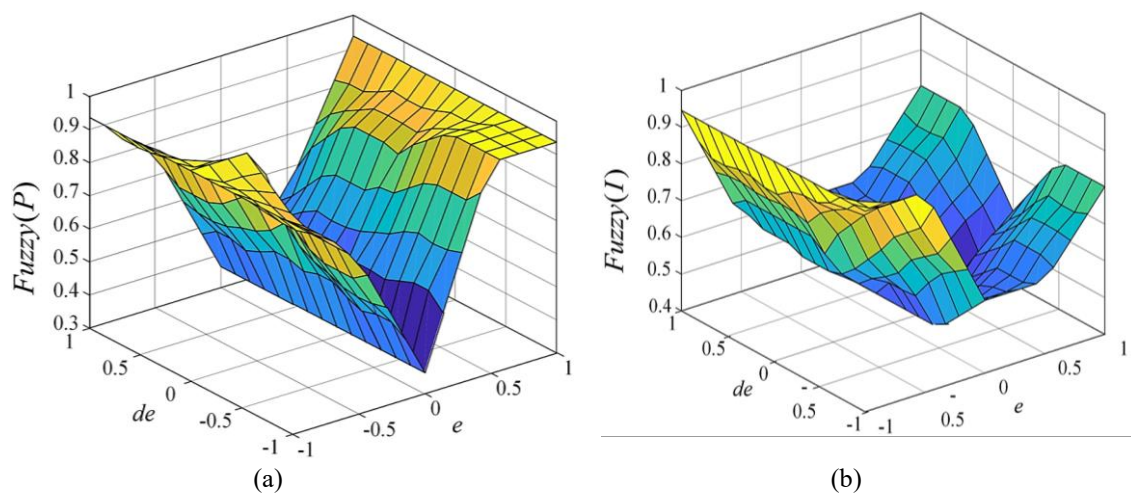


Figure 7. Control surfaces of fuzzy(P) and fuzzy(I): (a) fuzzy(P) surface and (b) fuzzy(I) surface

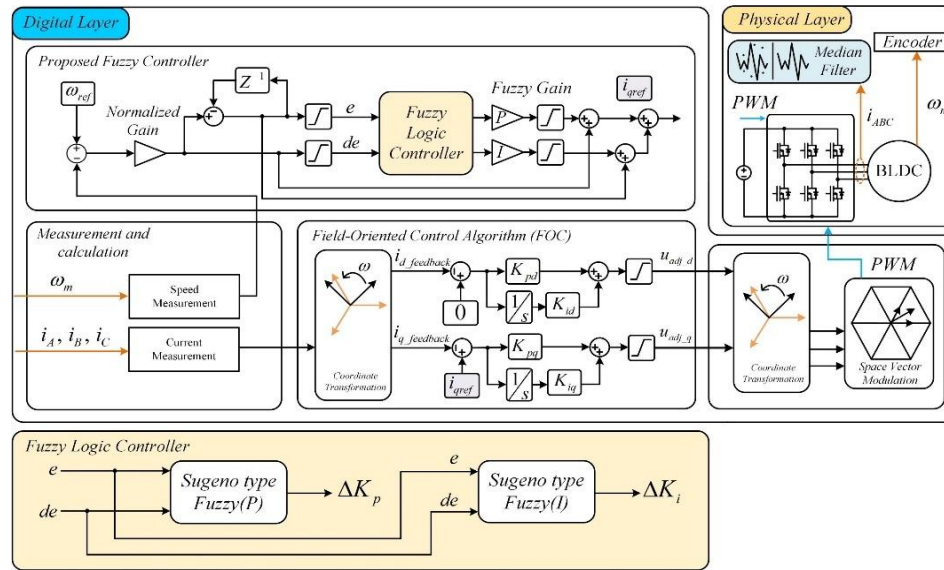


Figure 8. Proposed FOC control algorithm

5. SIMULATION RESULT ANALYSIS

In order to verify the effectiveness of the proposed fuzzy controller, the BLDC system is examined with parameters listed in Table 2. The conventional PI controller is used to compare with the proposed method to highlight the advantages of the separate P and I gain controller approach. The BLDC system is evaluated through scenarios with different speed control values. Figure 9 shows the control performance of the conventional and proposed Fuzzy logic controllers.

In Figures 9(a) and 9(b), the overall control performance ranges from 0 to 10s. In the first test scenario (Event 1), the speed reference is increased from 400 RPM to 1000 RPM. The conventional PI controller shows an overshooting percentage (OP) of 14.4% with a response time of 2.3 seconds. In contrast, the proposed fuzzy controller has a reduced overshoot of 8.5% with the same response time of 2.3 seconds.

The second scenario further investigates the controller behavior during a more substantial speed increase, from 1000 RPM to 2000 RPM, as shown in Figures 9(c) and 9(d). This scenario simulates an aggressive acceleration demand. Under these conditions, the limitations of the conventional PI controller become more apparent with overshoot of 12.1% and rise time of 0.54 seconds. The high overshoot shows that the fixed gains of the PI controller can not adapt the changing dynamics of the system. In contrast, the proposed fuzzy controller reduces the overshoot to 6.7% with rise time of 0.54 seconds. This result shows the adaptability of the proposed control approach, which adjusting its control parameters to minimize overshoot.

The third event examines the controller response to a decrease in the reference speed, from 2000 RPM to 1600 RPM, as shown in Figures 9(e) and 9(f). The conventional PI controller has an undershoot percentage (UP) of 6.06% with a fall time of 0.54 seconds. The proposed fuzzy controller shows a smaller undershoot of 3.94% but a longer fall time of 0.87 seconds. The simulation results indicate that the proposed controller has better control performance than that of the conventional controller, particularly in reducing overshoot and undershoot during speed reference changes.

Table 2. System and control parameters

Parameter category	Parameter	Value	Parameter	Value
Motor specifications	Stator resistance (R_s)	0485 Ω	Pole pairs ()	4
	Stator inductance (L_s)	0.0085 (H)	Rated voltage (V)	100 V
	Rotational inertia (J)	0.0027 kg.m ²	Rated speed (ω)	2500r/min
	Viscous damping (F)	0.000492 (N.m.s)		
Field-oriented control parameters	d-axis P gain (K_{pd})	20	q-axis P gain (K_{pq})	20
	d-axis I gain (K_{id})	40	q-axis I gain (K_{iq})	30
Conventional PI controller parameters	Gain P	0.05	gain I	0.5
Proposed fuzzy controller parameters	Normalized gain	0.00067	Fuzzy gain P	0.065
	Saturation normalized gain	-1 to 1	Fuzzy gain I	0.6
	Saturation fuzzy gain	0 to 1		

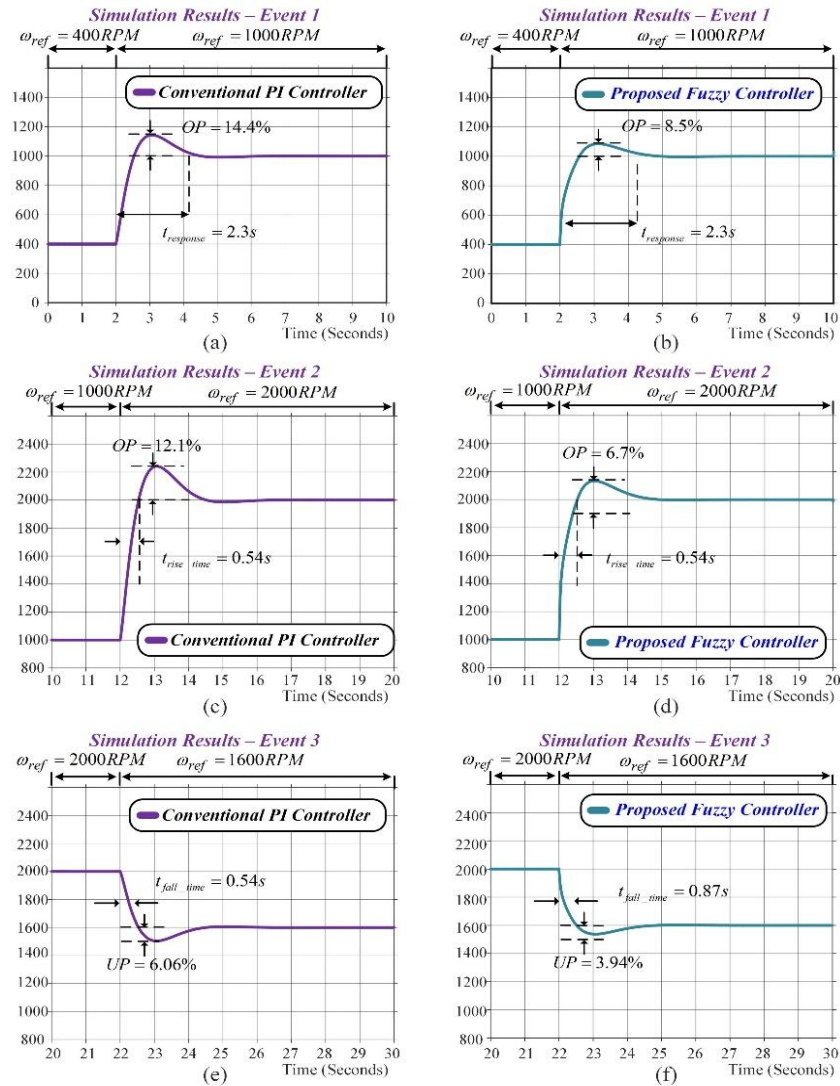


Figure 9. Control performance comparison of the proposed enhanced FLC and conventional PI controller
 (a)-(b) Even 1 with 1000 RPM speed reference; (c)-(d) Even 2 with speed increases from 1000 to 2000 RPM;
 (e)-(f) Event 3 when speed decreases to 1600RPM

6. EXPERIMENTAL SETUP AND RESULT ANALYSIS FOR BLDC SYSTEM

6.1. BLDC motor specifications and motor parameter measurement

In order to implement the experimental section, we use the BLDC Motor with ID “57BL75S10-230”. Because manufacturing tolerances often cause deviations from nominal values provided in datasheets, some measurement techniques are needed to measure the motor parameters. The motor parameters listed in Table 3 were established through experimental measurement using an LCR meter and other standard tools.

Table 3. Experimental BLDC motor parameters

Motor parameters	Value
Number of pole pairs	2 pole pairs
Rated speed	3000 RPM
Rated current	5.90 A
Rated voltage	24 V
Phase resistance	$0.42 \pm 10\% \Omega$
Phase inductance	$1.12 \pm 20\% mH$
Back EMF constant	4.3 Vrms/krpm
Encoder	-
Hall	120 degree

The following procedure was followed to measure the motor parameters, based on the guidelines provided in Infineon's application note on motor parameter measurement [27]. The phase-to-phase resistance (R_{l-l}) was measured using an LCR meter by connecting two motor terminals while leaving the third terminal open. The measured resistance was divided by two to obtain the phase resistance ($R_{phase} = R_{l-l}/2$). Similarly, the phase-to-phase inductance (L_{l-l}) was measured at a frequency of 1 kHz using the LCR meter across two motor terminals for phase inductance measurement. Then the measured inductance is divided values by 2. For motor back EMF constant, connect two phases to an oscilloscope voltage probe, leaving the third phase open. Then rotate the motor manually and measure the frequency and peak-to-peak voltage of the sinusoidal waveform. The formula to calculate the BEMF constant at 1000 RPM:

$$K_e = \frac{E_U 1000}{RPM} \quad (19)$$

where E_U and RPM are expressed by: $RPM = \frac{120}{p} \frac{1}{T_{elec}}$; $E_U = \frac{E_{pk}}{\sqrt{2}} = \frac{E_{pk}}{\sqrt{6}}$; where p is pole number, T_{elec} is the electrical angle period of time, E_{pk} is the peak line-to-line voltage (amplitude). This guide is particularly useful for engineers working on brushless DC motors for designing the motor controller.

6.2. Hall sensor calibration

Hall sensors are widely used in industrial applications because they are cost-effective, reliable, and robust. Their contactless operation eliminates wear and tear, making them durable in harsh environments with vibrations, extreme temperatures, and corrosive substances. Hall sensor calibration is critical for FOC because it ensures the accurate alignment of the rotor direct axis (d-axis) with the position detected by Hall sensors. This alignment is necessary for precise motor control, as FOC relies on knowing the rotor position to optimally energize the motor phases. Without proper calibration, the motor may experience inefficiencies or fail to operate correctly.

Manufacturers install Hall sensors near the stator of motors at specific angular intervals. Figure 10 shows the arrangement of three Hall sensors (a, b, and c) positioned 120 degrees apart around the stator of a three-phase motor. Hall sensors produce a three-bit output corresponding to their states: Hall A, B, and C. Each state represents a unique combination of high (1) or low (0) signals from the sensors. There are six valid states in an electrical cycle: 6 (110), 4 (100), 5 (101), 1 (001), 3 (011), and 2 (010). This sequence corresponds to the rotor movement through 60-degree commutation intervals. The predictable order ensures smooth motor operation by defining which phases should be energized at each step.

The calibration process involves determining the offset between the rotor d-axis and the position detected by Hall sensors. Firstly, run the motor in open-loop mode using constant voltage along the d-axis (V_d) and zero voltage along the q-axis (V_q). Operate the motor at low speed (50 RPM) using a ramp generator until its position reaches zero, so rotor position can be recorded as the offset value for Hall sensors. In the offset calib, d-axis voltage is fixed at 0.5 ($V_d = 0.5$) and $V_q = 0$ in order to ensures the rotor aligns with the d-axis. Then, the motor runs at a low speed (50 RPM) using a ramp generator, and the reference electrical angle is calculated through.

$$\begin{aligned} \omega_{elec_low_speed} &= \omega_{mech_low_speed_rad} pole_pairs = \omega_{mech_low_speed_RPM} \frac{2\pi}{60} pole_pairs \\ \theta_{elec_ref} &= \int \omega_{elec_low_speed} dt = \int \omega_{mech_low_speed_rad} pole_pairs \end{aligned} \quad (20)$$

Initially, the Hall sensor position ($\theta_{hall_measure}$) is calculated based on the current Hall states, and Hall state 6 is assumed to be at 0 radians before calibration. The offset is calculated by comparing the measured Hall position with the reference position:

$$\theta_{hall_offset} = \theta_{elec_ref} - \theta_{hall_measure}$$

Once calculated, this offset is used to correct future Hall sensor measurements:

$$\theta_{hall_measure_offset} = (\theta_{hall_measure} + \theta_{hall_offset})(mod \ (2\pi))$$

Figure 11 shows the experimental results of Hall sensor state sequences during motor operation. the six distinct Hall states (6→5→4→3→2→1) follow the expected binary pattern (110→101→100→011→010→001) in the correct sequence, confirming proper sensor functionality and

installation. Each state transition appears at approximately 60° electrical intervals, validating the theoretical positioning of the Hall sensors at 120° physical intervals around the stator.

Figures 12(a) and 12(b) show the experimental results of the measured and reference theta values before calibration. In Figure 12(a), we observe the comparison between the Hall-sensor-measured position ($\theta_{hall_measure}$) and the reference electrical angle (θ_{elec_ref}), with the calculated offset (θ_{hall_offset}) is clearly visible. The zoomed-in waveform in Figure 12(b) shows a detailed view of a single transition period. The reference electrical angle does not align with Hall state 3, which reveals manufacturing tolerance issues in Hall sensor positioning. The difference between measured and reference positions demonstrates why the Hall sensor offset calibration presented in this section is necessary. This calibration compensates for manufacturing variations to ensure proper motor control.

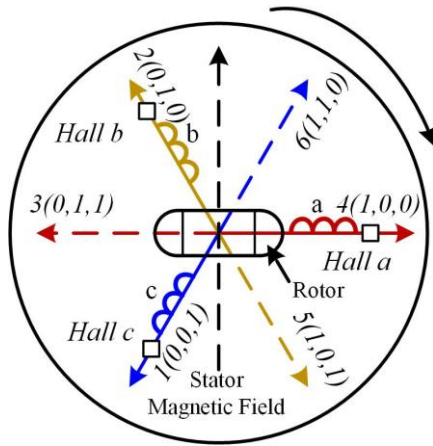


Figure 10. The hall state sequence and hall location

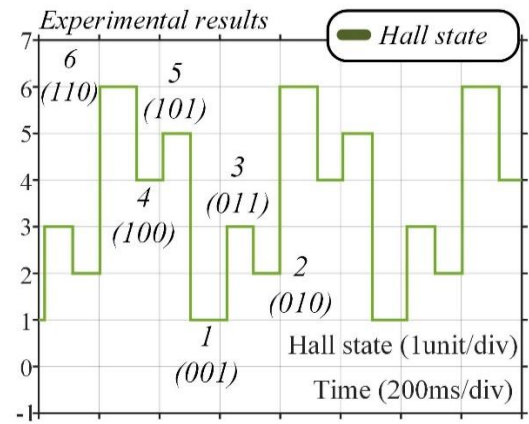


Figure 11. The experimental results of hall state sequence

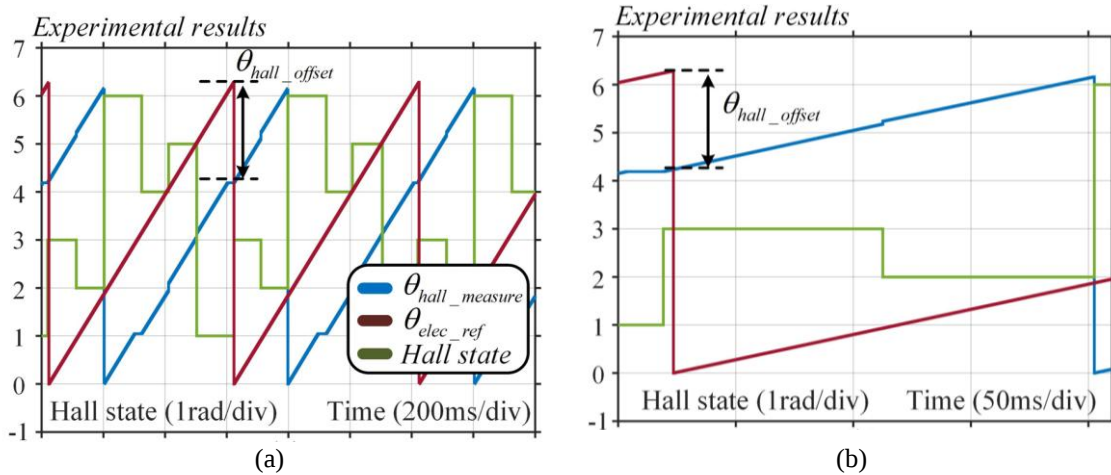


Figure 12. Experimental results of the electrical angle with Hall-sensor offset compensation: (a) the theta value reference and measurement and (b) the zoomed-in waveform

6.3. Experimental results and comparative analysis

In order to verify the effectiveness of the proposed FLC controller in practical application, an experimental prototype is constructed as shown in Figure 13. The Texas Instruments TMS320F28379D microprocessor is used to program the control algorithm using parameters in Table 2. The microcontroller is connected to the motor inverter driver board via a signal cable. In order to evaluate the control performance of the proposed fuzzy logic controller, experimental scenarios with different reference speeds are investigated, and the experimental results are compared with the conventional PI controller. In the first experimental scenario, the reference speed is increased from 400 to 1000 RPM, and the experimental results

are shown in Figures 14(a) and 14(b). Specifically, the conventional controller exhibits an overshooting percentage or OP of 9.44% with a response time of 2.18 s, while the proposed fuzzy controller achieves a significantly reduced OP of only 3.55% with the same response time.

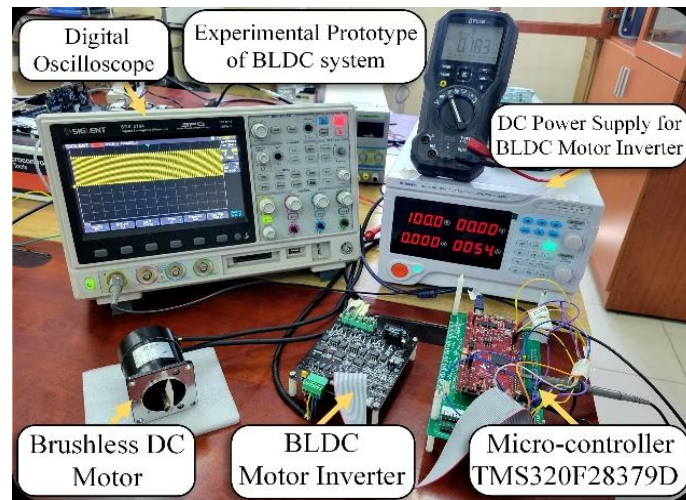


Figure 13. Experimental prototype system

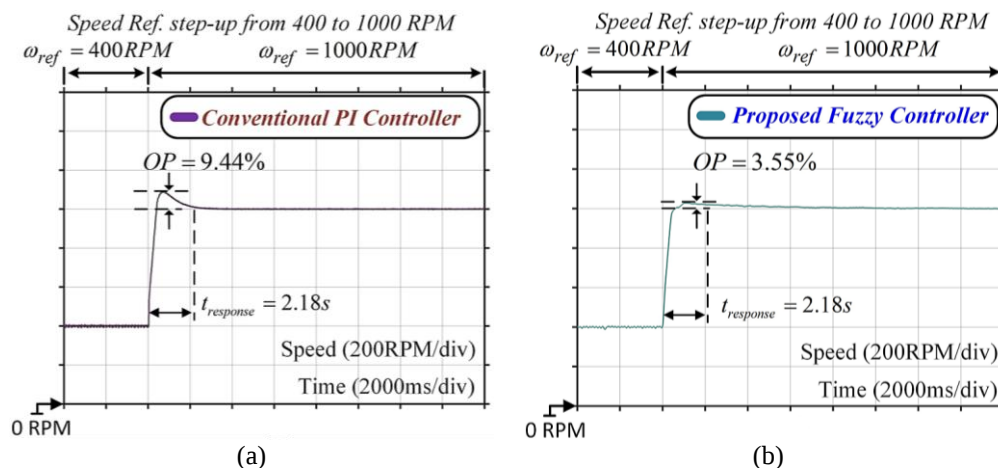


Figure 14. Experimental results when speed reference step-up from 400 to 1000 RPM:
(a) conventional PI controller and (b) proposed fuzzy controller

In the second experimental scenario, the reference speed is increased from 1000 to 2000 RPM, and the experimental results are shown in Figures 15(a) and 15(b). In this experimental scenario, the difference becomes more obvious as the conventional PI controller shows a significant OP of 33.33% with a rise time of 0.7 s, compared to the much lower OP of the proposed fuzzy controller of 8.57% with a slightly longer time of 0.77 s, as shown in Figures 15(a) and 15(b). In the scenario of reducing the reference speed from 2000 RPM to 1600 RPM, the control performance is shown in Figures 16(a) and 16(b). The conventional PI controller has an undershooting percentage (UP) of 15.28% with a downshift time of 0.77 s, while the proposed fuzzy controller still maintains superior performance with an UP of only 5.22% and a fall time of 0.83 s. From the experimental results in Figure 14 to Figure 16, the proposed controller shows the feasibility in achieving smoother speed transitions with significantly reduced overshoot and undershoot despite sometimes slightly longer response times. Comprehensive experimental testing validates both the effectiveness and practical viability of the proposed controller for practical BLDC motor speed control.

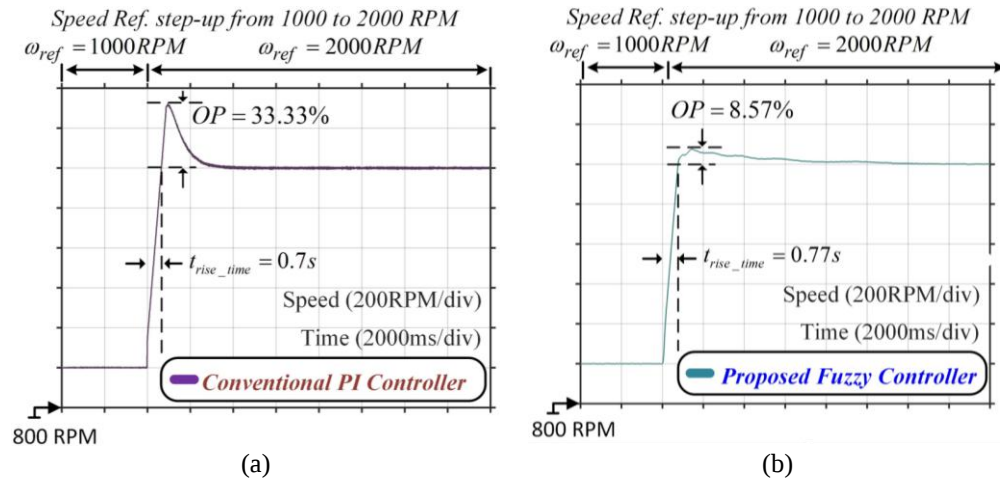


Figure 15. Experimental results when speed reference step-up from 1000 to 2000RPM:
(a) conventional PI controller and (b) proposed fuzzy controller

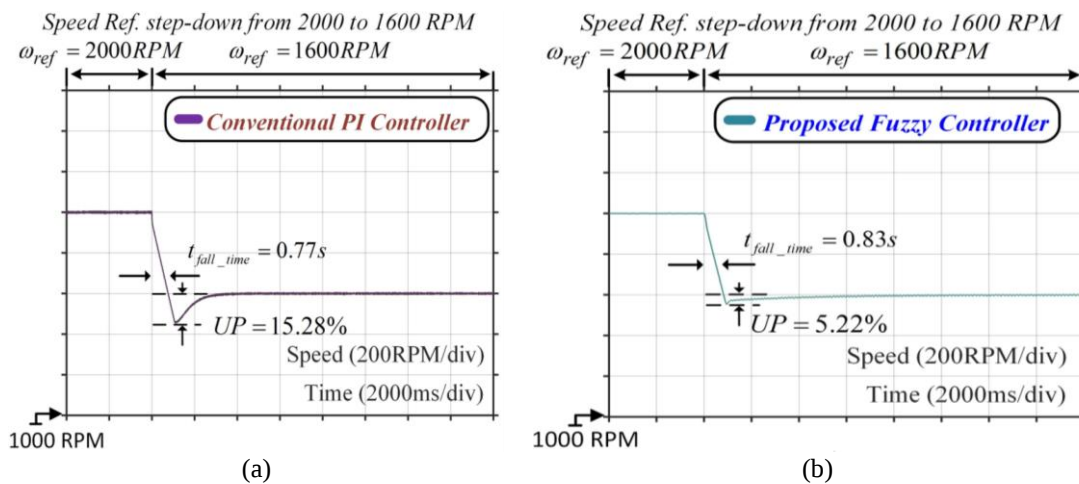


Figure 16. Experimental results when speed reference step-down from 2000 to 1600RPM:
(a) conventional PI controller and (b) proposed fuzzy controller

7. CONCLUSION

The paper presents a proposed fuzzy-PI control scheme with a median filter for enhancing BLDC motor speed regulation. The control approach uses two fuzzy logic controllers to adaptively adjust proportional and integral gains, addressing the limitations of conventional PI controllers. The experimental system is set up using a BLDC motor with ID "57BL75S10-230". The control algorithm is programmed using the Texas Instruments TMS320F28379D microcontroller. The experimental results show the performance and feasibility of the proposed method. The main results include a significant reduction in overshoots. Specifically, 3.55% vs. 9.44% in the 1st scenario, and 8.57% vs 33.33% in the 2nd scenario. The undershoot percentage is also reduced (5.22% < 15.28%) compared to the conventional PI controller, while maintaining the same or slightly longer response time. Comprehensive testing with various speed scenarios has validated the proposed controller's feasibility and reliability for BLDC motor applications.

ACKNOWLEDGEMENTS

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Minh Duc Pham	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	
Duong Nguyen Trong Qui			✓	✓		✓		✓	✓	✓	✓			
Truong Phuoc Hoa	✓	✓			✓		✓		✓	✓	✓	✓		✓

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article.

REFERENCES

- [1] P. Pillay and R. Krishnan, "Modeling, simulation, and analysis of permanent-magnet motor drives. ii. the brushless dc motor drive," *IEEE Transactions on Industry Applications*, vol. 25, no. 2, pp. 274–279, 1989, doi: 10.1109/28.25542.
- [2] D. Mohanraj *et al.*, "A review of BLDC motor: state of art, advanced control techniques, and applications," *IEEE Access*, vol. 10, pp. 54833–54869, 2022, doi: 10.1109/ACCESS.2022.3175011.
- [3] A. Eldho Aliasand and F. T. Josh, "Selection of motor foran electric vehicle: a review," *Materials Today: Proceedings*, vol. 24, pp. 1804–1815, 2020, doi: 10.1016/j.matpr.2020.03.605.
- [4] J. E. Muralidhar and P. V. Aranasi, "Torque ripple minimization & closed loop speed control of BLDC motor with hysteresis current controller," in *Proceedings of the IEEE International Caracas Conference on Devices, Circuits and Systems, ICCDCS*, IEEE, Mar. 2014, pp. 1–7, doi: 10.1109/ICDCSyst.2014.6926122.
- [5] N. Li, M. Li, C. Zhang, X. Niu, and Z. Wang, "Design of brushless motor control system based on foc algorithm," in *International Conference on Mechatronic Engineering and Artificial Intelligence (MEAI 2023)*, F. Liu and Y. Wei, Eds., SPIE, Feb. 2024, p. 183, doi: 10.1117/12.3025713.
- [6] M. Akrami, E. Jamshidpour, B. Nahid-Mobarakeh, S. Pierfederici, and V. Frick, "Sensorless control methods for BLDC motor drives: a review," *IEEE Transactions on Transportation Electrification*, vol. 11, no. 1, pp. 135–152, Feb. 2025, doi: 10.1109/TTE.2024.3387371.
- [7] N. S. Nise, "Control systems engineering, 8th edition | Wiley." pp. 1–800, 2020. [Online]. Available: <https://www.wiley.com/en-us/Control+Systems+Engineering%2C+8th+Edition-p-9781119474227>.
- [8] A. Usman and B. S. Rajpurohit, "Design and control of a BLDC motor drive using hybrid modeling technique and FPGA-based hysteresis current controller," in *2020 IEEE 9th Power India International Conference (PIICON)*, IEEE, Feb. 2020, pp. 1–5, doi: 10.1109/PIICON49524.2020.9112895.
- [9] Y. Wakasa, S. Kanagawa, K. Tanaka, and Y. Nishimura, "Controller parameter tuning for systems with hysteresis and its application to shape memory alloy actuators," *SICE Journal of Control, Measurement, and System Integration*, vol. 5, no. 3, pp. 162–168, May 2012, doi: 10.9746/jcmsi.5.162.
- [10] M. Mahmud, M. R. Islam, S. M. A. Motakabber, M. D. A. Satter, K. E. Afroz, and A. K. M. Ahasan Habib, "Control speed of bldc motor using pid," in *2022 IEEE 18th International Colloquium on Signal Processing and Applications, CSPA 2022 - Proceeding*, IEEE, May 2022, pp. 150–154, doi: 10.1109/CSPA55076.2022.9782030.
- [11] A. G. Daful, "Comparative study of pid tuning methods for processes with large & small delay times," in *2018 Advances in Science and Engineering Technology International Conferences (ASET)*, IEEE, Feb. 2018, pp. 1–7, doi: 10.1109/ICASET.2018.8376915.
- [12] R. Anandanatarajan, M. Chidambaram, and T. Jayasingh, "Limitations of a pi controller for a first-order nonlinear process with dead time," *ISA Transactions*, vol. 45, no. 2, pp. 185–199, Apr. 2006, doi: 10.1016/S0019-0578(07)60189-X.
- [13] Y.-H. Yoon and J.-M. Kim, "Precision control of a sensorless PM BLDC motor using PLL control algorithm," *Electrical Engineering*, vol. 100, no. 2, pp. 1097–1111, Jun. 2018, doi: 10.1007/s00202-017-0571-x.
- [14] S. Kumar, D. Roy, and M. Singh, "A fuzzy logic controller based brushless dc motor using pfc cuk converter," *International Journal of Power Electronics and Drive Systems*, vol. 10, no. 4, pp. 1894–1905, Dec. 2019, doi: 10.11591/ijpeds.v10.i4.pp1894-1905.
- [15] M. A. Ibrahim, A. K. Mahmood, and N. S. Sultan, "Optimal pid controller of a brushless dc motor using genetic algorithm," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 10, no. 2, p. 822, Jun. 2019, doi: 10.11591/ijpeds.v10.i2.pp822-830.
- [16] K. S. R. Rao and A. H. Bin Othman, "Design optimization of a bldc motor by genetic algorithm and simulated annealing," in *2007 International Conference on Intelligent and Advanced Systems, ICIAS 2007*, IEEE, Nov. 2007, pp. 854–858, doi: 10.1109/ICIAS.2007.4658508.

- [17] E. H. Putra, Z. Has, and M. Effendy, "Robust adaptive sliding mode control design with genetic algorithm for brushless dc motor," in *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, IEEE, Oct. 2018, pp. 330–335. doi: 10.1109/EECSI.2018.8752768.
- [18] K. Ullah, J. Guzinski, and A. F. Mirza, "Critical review on robust speed control techniques for permanent magnet synchronous motor (pmsm) speed regulation," *Energies*, vol. 15, no. 3, p. 1235, Feb. 2022, doi: 10.3390/en15031235.
- [19] V. K. Karan, S. Shekhar, A. Alam, and A. Thakur, "Elimination of torque ripples by multiple slope ST-DTC vectors in PM-BLDC drive," *Electrical Engineering*, vol. 106, no. 3, pp. 3393–3402, Jun. 2024, doi: 10.1007/s00202-023-02155-0.
- [20] J. Shi, Q. Mi, W. Cao, and L. Zhou, "Optimizing BLDC motor drive performance using particle swarm algorithm-tuned fuzzy logic controller," *SN Applied Sciences*, vol. 4, no. 11, p. 293, Nov. 2022, doi: 10.1007/s42452-022-05179-6.
- [21] J. Rodriguez *et al.*, "Latest advances of model predictive control in electrical drives—part ii: applications and benchmarking with classical control methods," *IEEE Transactions on Power Electronics*, vol. 37, no. 5, pp. 5047–5061, May 2022, doi: 10.1109/TPEL.2021.3121589.
- [22] S. Mohammad Taher, A. Halvaei Niasar, and S. Abbas Taher, "A new mpc-based approach for torque ripple reduction in bldc motor drive," in *2021 12th Power Electronics, Drive Systems, and Technologies Conference, PEDSTC 2021*, IEEE, Feb. 2021, pp. 1–6. doi: 10.1109/PEDSTC52094.2021.9405871.
- [23] Q. Wang, J. Cao, and H. Liu, "Adaptive fuzzy control of nonlinear systems with predefined time and accuracy," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 12, pp. 5152–5165, Dec. 2022, doi: 10.1109/TFUZZ.2022.3169852.
- [24] C. L. Xia, "Permanent magnet brushless dc motor drives and controls," *Permanent Magnet Brushless DC Motor Drives and Controls*, 2012, doi: 10.1002/9781118188347.
- [25] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff, "Analysis of electric machinery and drive systems," *Analysis of Electric Machinery and Drive Systems*, 2010, doi: 10.1109/9780470544167.
- [26] E. Nandhini and A. Sivaprakasam, "A review of various control strategies based on space vector pulse width modulation for the voltage source inverter," *IETE Journal of Research*, vol. 68, no. 5, pp. 3187–3201, 2022, doi: 10.1080/03772063.2020.1754935.
- [27] I. T. A. G, "Basic motor parameters and the configuration," Infineon Technologies AG, Munich, Germany, 2023. [Online]. Available: https://www.infineon.com/dgdl/Infineon-Basic_Motor_Parameters_and_the_Configuration-ApplicationNotes-v01_00-EN.pdf?fileId=8ac78c8c8caa022e018cee2157f0686d

BIOGRAPHIES OF AUTHORS



Minh Duc Pham    received the master and Ph.D. degrees in Electrical Engineering from Ulsan University, South Korea. He is currently a full-time lecturer at Ho Chi Minh City University of Technology, Vietnam. His research interests include hybrid AC-DC microgrids, PMSM, low-cost inverters, and renewable energy. He can be contacted at email: pmduc@hcmut.edu.vn.



Duong Nguyen Trong Qui    is a researcher at the Faculty of Electrical and Electronic Engineering, Ho Chi Minh City University of Technology, Vietnam. His research interests include control theory, PMSM, embedded systems, and power electronics. He can be contacted at email: duong.nguyentrong2801@hcmut.edu.vn.



Truong Phuoc Hoa    received the B.E. (2006), M.E. (2009) degrees in electrical engineering from Ho Chi Minh City University of Technology, Vietnam, and the Ph.D. degree in electrical and electronic engineering from University of Haute Alsace, Mulhouse, France, in 2016. His research interests include the design and control of synchronous motors. He can be contacted at email: tphoa@hcmut.edu.vn.